



TITLE:

Recursion opertor[operator], Hereditary operator 及び Schouten bracket の計算について(数式処理 と数学研究への応用)

AUTHOR(S):

渡辺, 芳英; 加古, 富志雄

CITATION:

渡辺, 芳英 ...[et al]. Recursion opertor[operator], Hereditary operator 及び Schouten bracket の計算について(数式処理と数学研究への応用). 数理解析研究所講究録 1990, 722: 50-58

ISSUE DATE:

1990-05

URL:

<http://hdl.handle.net/2433/101847>

RIGHT:

Recursion operator, Hereditary operator 及び Schouten bracket の計算について

広島大学工学部 渡辺芳英 (YOSHIHIDE WATANABE)
広島大学理学部 加古富志雄 (FUJIO KAKO)

1. 予備概念.

A. Lie-Bäcklund symmetry と保存量.

$R(u)$ は u_0, u_1, u_2, \dots の実数係数の有理関数体で $Du_j = u_{j+1}$ ($j = 0, 1, 2, \dots$) をみたす微分 D をもつものとする。習慣に従って u_0 を u とかき、微分 $\partial/\partial u_i$ ($i = 0, 1, 2, \dots$) を ∂_i と略記する。 $R(u)$ の元 f の発展微分 ∂_f を

$$(1.1) \quad \partial_f = \sum_{j \geq 0} (D^j f \partial_j),$$

で定義する。今 $R(u)$ にブラケット $[,]$ を

$$(1.2) \quad [f, g] = \partial_f g - \partial_g f.$$

で導入すれば、発展微分は以下の交換関係をみたす:

$$(1.3) \quad [\partial_f, \partial_g] = \partial_f \partial_g - \partial_g \partial_f = \partial_{[f, g]}.$$

$u(x, t)$ についての発展方程式

$$(1.4) \quad u_t = H(u, u_1, u_2, \dots, u_m) \quad (H \in R(u)),$$

を考える。ここで u_t は $(\partial/\partial t)u(x, t)$ を意味し、 u_j は $(\partial/\partial x)^j u(x, t)$ を意味するものとする。このような発展方程式を考えたとき ∂_H は方程式上での時間微分になる。 $R(u)$ 上に微分作用素:

$$(1.5) \quad D(H) = \sum_{j \geq 0} (\partial_j H) D^j$$

を導入して、微分方程式

$$(1.6) \quad (\partial_H - D(H))f = 0,$$

を考えよう。この微分方程式 (1.6) の ($R(u)$ の適当な拡大体での) 解を方程式 (1.4) の Lie-Bäcklund symmetry (以後単に symmetry) とよぶ。 f が方程式 (1.4) の symmetry なら方程式の解 u に対して無限小変換 $u + \epsilon f$ は ϵ の高次のオーダーを無視すれば方程式の解であり、それが symmetry の最も直観的な意味である。(1.2) で定義したブラケットを用いて定義方程式 (1.6) を書き直せば $[H, f] = 0$ となり、方程式 (1.4) の symmetry 全体 L_H は (1.2) で定義される Lie ブラケットをもつ Lie 環となる。方程式 (1.4) は独立

変数 x, t を陽に含んでいないので x, t に関する平行移動にたいして不変であり、その事実に応じて u_1, H という二つの symmetry を常にもつ (自明な symmetry とよばれる)。

KdV 方程式等のいわゆる soliton 方程式においては L_H は無限次元可換 Lie 環になるかまたはそのような Lie 環を部分 Lie 環として含んでおり、無限個の生成元は recursion operator とよばれる D に関する微分積分作用素を自明な symmetry に順次作用させることによって得られることが知られている ([O])。 $R(u)$ 上の微分積分作用素 \mathcal{D} (一般には $R(u)$ 係数の D の Laurent 級数で与えられる) が $\mathcal{D}L_H \subset L_H$ をみたすとき \mathcal{D} は (symmetry にたいする) recursion operator であるとよばれる。容易にわかるように、 \mathcal{D} が symmetry を定義する微分作用素と可換、すなわち

$$(1.7) \quad [\partial_H - D(H), \mathcal{D}] = 0,$$

ならば \mathcal{D} は recursion operator となる (Olver [O])。 Recursion operator を自明な symmetry H に順次作用させて無限個の symmetry $\{D^n H; n = 1, 2, 3, \dots\}$ が得られるが、それらが可換であるためには \mathcal{D} が任意の $f \in R(u)$ にたいして

$$(1.8) \quad \mathcal{D} \cdot [\mathcal{D}, \partial_f - D(f)] = [\mathcal{D}, \partial_{\mathcal{D}f} - D(\mathcal{D}f)]$$

をみたせばよい。 Fuchssteiner ([F]) はこのような作用素を hereditary operator とよんだ。知られている recursion operator はすべて hereditary operator であり反例は知られていない。

次に発展方程式の保存密度と保存量について説明する。 $R(u)$ の元 f が方程式 (1.4) の保存密度であるとは

$$(1.9) \quad \partial_H f = Dg$$

をみたす $g \in R(u)$ が存在するときをいう。方程式の上では ∂_H は時間微分であり、 D は x 微分である。従って保存密度 f を適当な u にたいする境界条件 (たとえば $x \rightarrow \pm\infty$ で急減少) のもとで積分したものは時間によらない量である。積分の概念を代数化する為に次のような同値関係 \sim を $R(u)$ に導入する。すなわち、 $f, g \in R(u)$ にたいして $f - g = Dh$ となる $h \in R(u)$ が存在するとき f と g は同値であるといって $f \sim g$ とかく。その時 (1.9) は

$$(1.9') \quad \partial_H f \sim 0$$

と同値である。 $R(u)$ における同値類の集合を $\tilde{R}(u)$ と書き、 f の属する同値類 \tilde{f} を形式的な積分 $\int f dx$ で表す。方程式 (1.4) の保存密度の属する同値類を保存量とよぶことにする。

Soliton 方程式は無限個の保存量を持ち、それらの保存量を方程式の何らかの意味での対称性の概念と結び付けようとすれば、symmetry の概念が必要となる。そして保存量と symmetry を結び付ける枠組みが次に述べる Hamilton 構造の概念である。

B. Schouten bracket と Hamilton 構造.

まず、 $R(u)$ 上に $\tilde{R}(u)$ 値双線形形式 \langle, \rangle を

$$(1.10) \quad \langle f, g \rangle = \int f g dx \quad (f, g \in R(u))$$

で定義する。有限階の微分積分作用素

$$\mathcal{P} = \sum_{j=-\infty}^n p_j D^j \quad (p_j \in R(u) ; n : \text{整数})$$

の形式的共役作用素 \mathcal{P}^* を

$$\mathcal{P}^* = \sum_j (-1)^j D^j \cdot p_j = \sum_j \sum_{k \geq 0} (-1)^j (D^k p_j) \binom{j}{k} D^{j-k}.$$

で定めれば、

$$(1.11) \quad \langle \mathcal{P}f, g \rangle = \langle f, \mathcal{P}^*g \rangle$$

が $(\mathcal{P}f$ と \mathcal{P}^*g が well-defined であるような) 任意の $f, g \in R(u)$ についてなりたつ。

微分積分作用素 \mathcal{P} と \mathcal{Q} にたいしてその Schouten bracket $[[\mathcal{P}, \mathcal{Q}]]$ を $R(u)$ 上の $\tilde{R}(u)$ 値三重線形形式

$$(1.12) \quad [[\mathcal{P}, \mathcal{Q}]](f, g, h) = \langle \partial_{\mathcal{P}f}(\mathcal{Q})g, h \rangle + \langle \partial_{\mathcal{Q}f}(\mathcal{P})g, h \rangle \\ + \text{cyclic sum for } (f, g, h), \quad f, g, h \in R(u),$$

で定義する ([GD1,2])。ここで $\partial_f(\mathcal{P})(f \in R(u))$ は $\mathcal{P} = \sum p_j D^j$ の各係数を ∂_f で微分した作用素

$$\partial_f(\mathcal{P}) = \sum (\partial_f p_j) D^j$$

をあらわす。

\mathcal{H} を歪対称な、すなわち $\mathcal{H}^* = -\mathcal{H}$ をみたす作用素とする。 \mathcal{H} が任意の $f, g, h \in R(u)$ にたいして

$$(1.13) \quad [[\mathcal{H}, \mathcal{H}]](f, g, h) = 0$$

がなりたつとき \mathcal{H} は Hamilton 作用素であるとよばれる。 \mathcal{H} が Hamilton 作用素ならばそれを用いて $\tilde{R}(u)$ における Poisson bracket $\{, \}_{\mathcal{H}}$ を

$$(1.14) \quad \{\tilde{f}, \tilde{g}\}_{\mathcal{H}} = \left\langle \frac{\delta f}{\delta u}, \mathcal{H} \frac{\delta g}{\delta u} \right\rangle,$$

で定める。ここで $\delta/\delta u$ は

$$(1.15) \quad \frac{\delta}{\delta u} = \sum_j (-1)^j D^j \cdot \partial_j.$$

で定義される Euler operator である。 \mathcal{H} は歪対称であるから Poisson bracket (1.14) は歪対称であり、さらに (1.13) から Jacobi の恒等式をみたすことが導かれる。

\mathcal{H} と \mathcal{K} を Hamilton 作用素として、その Schouten bracket が 0 すなわち任意の $f, g, h \in R(u)$ について $[\mathcal{H}, \mathcal{K}](f, g, h) = 0$ となるとき $(\mathcal{H}, \mathcal{K})$ を Hamilton 対とよぶ。 $(\mathcal{H}, \mathcal{K})$ が Hamilton 対であればその定数係数の一次結合は Hamilton 作用素になる。また recursion operator \mathcal{D} が二つの Hamilton 作用素 \mathcal{H} と \mathcal{K} を用いて $\mathcal{D} = \mathcal{K} \cdot \mathcal{H}^{-1}$ と表され、さらに $(\mathcal{H}, \mathcal{K})$ が Hamilton 対をなせば \mathcal{D} は hereditary operator となることが知られている (Gel'fand-Dorfman [GD2], Fuchssteiner-Fokas [FF])。

発展方程式 $u_t = H$ ($H \in R(u)$) について H が適当な Hamilton 作用素 \mathcal{H} と保存密度 h を用いて

$$(1.16) \quad H = \mathcal{H} \frac{\delta h}{\delta u}$$

と表されるとき、発展方程式は Hamilton 形をもつという。発展方程式が Hamilton 形をもてば $\tilde{f} \in \tilde{R}(u)$ 保存量であるという条件は $\{\tilde{h}, \tilde{f}\}_{\mathcal{H}} = 0$ と同値であり、方程式の保存量全体は Lie 環 $(C_H, \{, \}_{\mathcal{H}})$ をなす。さらに $\mathcal{H} \frac{\delta}{\delta u}$ は $(C_H, \{, \}_{\mathcal{H}})$ から $(L_H, [,])$ への Lie 環としての準同形を引き起こすことがわかる。我々が興味をもつ発展方程式は recursion operator \mathcal{D} を持つ Hamilton 方程式 (Hamilton 作用素 \mathcal{H}) で \mathcal{D} が \mathcal{H} と Hamilton 対をなす Hamilton 作用素 \mathcal{K} を用いて $\mathcal{D} = \mathcal{K} \cdot \mathcal{H}^{-1}$ と書けるようなものである。そのような発展方程式は無数個の保存量を持ち、それらの保存量は Poisson bracket に関して可換 (対合的又は包含的ともいう) である ([FF], [GD2])。このような方程式を質点系の古典力学の類似から形式的完全可積分系と呼ぶことにする。

著者の内一人 (Y.W.) は共著論文 [FW1,2] において、自明でない symmetry を持つあるタイプの 3 階の発展方程式の分類を行い、それ等の方程式の殆どすべてに対して recursion operator を見出した。更に [W] においてそのうち幾つかの発展方程式について Hamilton 構造を見出し、それ等の方程式の recursion operator が Hamilton 対をなす二つの Hamilton 作用素の商でかけることを示すことによってそれ等の方程式の形式的完全可積分性を証明した。その計算は単純ではあるが極めて繁雑であり、その過程でもう一人の著者 (F.K.) の協力を得て幾つかのプログラムを作った。次節でそのうち幾つかを紹介する。

2. REDUCE でのプログラム.

A. REDUCE での表現について.

$R(u)$ の生成元 $u_0 (= u), u_1, u_2, \dots$ を表すために u をオペレーター宣言 (operator u ;) してそれらを $u(0), u(1), u(2), \dots$ で表す。 D による微分は x 関する微分として微分則 $Du_j = u_{j+1}$ ($j = 0, 1, 2, \dots$) を let 文を用いて定義する。この微分は非常に多く呼び出されるのでプログラムを高速化するためには専用の微分パッケージを作るかまたは REDUCE 自体の微分のパッケージを書き換えたほうがよい (実際のプログラムでは最終的に後者を選択した)。

$R(u)$ 係数の D に関する微分作用素はそのまま D に関する $R(u)$ 係数の多項式として表す。積分作用素としては $rD^{-1} \cdot s$, $r, s \in R(u)$ の形の作用素の有限和だけを考える。ここで作用素 $rD^{-1} \cdot s$ を $R(u)$ の元 f に作用させるとは sf を積分して r をかけることを意味するものとする (sf がある関数の微分でかけているとして)。作用素 $rD^{-1} \cdot s$ をオペレーター ddi (operator ddi ;) を用いて $r * ddi(s)$ で表す。例えば KdV 方程式 $u_t = u_3 + 3uu_1$

の recursion operator: $D^2 + 2u + u_1 D^{-1}$ は $D ** 2 + 2 * u(0) + u(1) * ddi(1)$ と表されるし Hamilton 作用素 $u_1 D^{-1} \cdot u_1$ は $u(1) * ddi(u(1))$ で表される。

与えられた作用素が hereditary operator であることを確かめたり、また作用素どうしの Schouten bracket を計算するには作用素を任意関数に作用させる必要がある。作用素の関数への作用は *diop* という procedure で実行されるが、作用される関数はオペレーター宣言され (例えば f に作用するなら operator f ;) 微分作用素 D^n を $f = f(0)$ に作用させた結果 $diop(D^n, f(0))$ は $f(n)$ となる。積分作用素については *ind* というオペレーター (operator *ind*; for all f let $df(ind(f), x) = f$;) を準備して、例えば $\mathcal{H} = u_1 D^{-1} \cdot u_1$ を f に作用させた結果 $diop(\mathcal{H}, f(0))$ を $u(1) * ind(u(1) * f(0))$ とする。ここで注意すべきは *ind* には線形性が仮定されていないことで、例えば $ind(2 * u(0) - 3 * u(1))$ はこのままでは $2 * ind(u(0)) - 3 * ind(u(1))$ と展開されず、実際は打ち消しあう項が消えなくなる。この問題を解決するためには *ind* を展開する procedure である *consind* 又は *expandind(consind(f))* は $ind(f)$ の展開形を求め、*expandind(f)* は f のなかの *ind* を含む項を展開する) を作って積分作用素を作用した後は必ずその procedure を起動するようにすればよい。有理式 f にたいして *ind(f)* を展開するには、まず f の分子を単項の和に分けることによって f を分子が単項である有理式の和に分け、分子からは定数部分を括りだし、分母からは共通定数因子を括りだすようにする。この部分のプログラムは実際に作ってみると意外に面倒であり、まだ不完全なところがあるかもしれない。なお、上記の展開を簡単に行うためだけなら、*ind* を linear operator と宣言し (linear *ind*;) 更に depend u, x ; として $ind(2 * u(0) - 3 * u(1), x)$ を評価すればよい。このことは最近伊藤雅明氏に教わった。

B. Recursion operator であることを確かめる。

微分積分作用素 \mathcal{D} が recursion operator となる条件 (1.7) を書き換えると、

$$(2.1) \quad \partial_H(\mathcal{D}) - [D(H), \mathcal{D}] = 0$$

となる。(3.1) の左辺を計算するには以下の三つの procedure を作ればよい。

- (1) 微分積分作用素 \mathcal{D} の各係数を ∂_H で微分した作用素 $\partial_H(\mathcal{D})$ を計算するための *dfev*(\mathcal{D}, H).
 - (2) 与えられた関数 H にたいして (1.5) で定義される微分作用素 $D(H)$ を構成する *fdiff*(H).
 - (3) 微分積分作用素 \mathcal{D} と微分作用素との交換子を計算するために必要な、作用素としての積を計算する *seki*(f, g) (但し f, g の一方は微分作用素で十分).
- (1) について積分作用素 $\mathcal{D} = f D^{-1} \cdot g$ にたいしては

$$\partial_H(\mathcal{D}) = \partial(f) D^{-1} \cdot g + f D^{-1} \cdot \partial_H(g)$$

であるから、対象となる作用素が積分作用素を含む場合は、前部分節 (2-A) の終わりに *ind* について述べた注意がこの場合 *ddi* についてあてはまる。例えば $\mathcal{D} = ind(u(0))$ で $H = u(3) + 3 * u(1) * u(0)$ (KdV 方程式) とすれば $dfev(\mathcal{D}, H) = ddi(u(3) + 3 * u(1) * u(0))$ となるが、この *ddi* を展開しなければ計算は進まない (その為の procedure は *consddi*)。 (2) の procedure *fdiff* を作るのは簡単で説明を要しないであろう。

(3) の作用素としての積 (一方は微分作用素) を計算する場合はまず対象となる作用素を微分作用素の部分と積分作用素の部分の和に分けて計算することにすれば、以下の四つの procedure をつくればよい。

(3-1) 微分積分作用素 \mathcal{P} が与えられたとき、その微分作用素部分と積分作用素部分を求める procedure $dipart(\mathcal{P})$ 。

(3-2) 微分作用素 \mathcal{P} と \mathcal{Q} の積を計算する procedure $sekidd(\mathcal{P}, \mathcal{Q})$ 。

(3-3) 微分作用素 \mathcal{P} と積分作用素 \mathcal{D} の積を計算する procedure $sekidi(\mathcal{P}, \mathcal{D})$ 。

(3-4) 積分作用素 \mathcal{D} と微分作用素 \mathcal{P} の積を計算する procedure $sekiid(\mathcal{D}, \mathcal{P})$ 。

$sekidd$ を作るには $seki(D, \mathcal{P}) = D \cdot \mathcal{P} = \mathcal{P} \cdot D + df(\mathcal{P}, x)$ であることに注意すればよい。 $sekidi$ を作るには、微分作用素 \mathcal{P} と積分作用素 $f \cdot ddi(g) = fD^{-1} \cdot g$ にたいして、まず \mathcal{P} を定数項 \mathcal{P}_0 とその残り \mathcal{P}_1 (\mathcal{P}_1 は D で割り切れる) に分け、公式

$$seki(\mathcal{P}, f * ddi(g)) = \mathcal{P} \cdot fD^{-1} \cdot g = \frac{\mathcal{P}_1}{D} \cdot fg + (D(f)g + \mathcal{P}_0 f)D^{-1} \cdot g$$

を繰り返し (再帰的に) 用いればよい。 $sekiid$ を作るには公式

$$D^{-1} \cdot h \cdot D = D^{-1} \cdot (D \cdot h - D(h)) = h - D^{-1} \cdot D(h)$$

を繰り返し (再帰的に) 用いる。

C. Hereditary operator であることを確かめる。

微分積分作用素 \mathcal{D} が hereditary operator となる条件 (1.8) を $R(u)$ の元 g に作用した式を発展微分を用いて書き直すと

$$(2.2) \quad (\partial_{\mathcal{D}f}(\mathcal{D}) - \mathcal{D} \cdot \partial_f(\mathcal{D}))g - (\partial_{\mathcal{D}g}(\mathcal{D}) - \mathcal{D} \cdot \partial_g(\mathcal{D}))f = 0$$

となり、(2.2) が任意の $f, g \in R(u)$ について成り立てば \mathcal{D} は hereditary operator となることがわかる。(2.2) の左辺を計算するためにあらたに以下の二つの procedure を作る。

(1) 微分積分作用素 \mathcal{D} を $R(u)$ の元 f に作用させる $diop(\mathcal{D}, f)$ 。このとき部分節 (2-A) で注意したように積分作用素を作用させた場合に生じる ind の引数はすべて展開しなければならない。

(2) (2.2) 式の左辺の第一項と第二項は f と g を入れ替えたものにすぎないので、第一項だけを計算して (それでもかなりの時間がかかる場合が多い) 第二項は第一項の計算結果の式で f と g を入れ替える。そのための $conv(p, f, g)$ (有理式 p のなかに含まれるオペレーター名 f と g を入れ替える procedure)。

以上の準備のもとで微分積分作用素 \mathcal{D} にたいして (2.2) 式の左辺を計算する procedure $herop(\mathcal{D})$ を作ることが出来る。但し REDUCE ではオペレーター名 f, g は大域的になってしまうのでこの procedure を呼ぶ前に f, g を変数名などに使うとエラーがおきる。例として $\mathcal{D} = D^2 + 2u + u_1 D^{-1}$ (KdV 方程式の recursion operator) に $herop$ を適用すると結果は

$$(2.3) \quad u(1) * (f(0) * ind(g(0)) - g(0) * ind(f(0))) \\ - ind(f(1) * ind(g(0))) + ind(g(1) * ind(f(0)))$$

となって 0 とならない。しかし部分積分の公式

$$(2.4) \quad \begin{aligned} \text{ind}(f(1) * \text{ind}(g(0))) &= \text{ind}(df(f(0) * \text{ind}(g(0)), x) - f(0) * g(0)) \\ &= f(0) * \text{ind}(g(0)) - \text{ind}(f(0) * g(0)) \end{aligned}$$

及びその類似公式を用いれば (2.3) は 0 となる。このように $\text{herop}(\mathcal{D})$ を実行した結果は \mathcal{D} が hereditary operator であっても通常は 0 にならず、0 とするためには (2.4) のような部分積分が必要である。どのようなアルゴリズムで部分積分を行ったらよいかについてはまだよくわからないが、例えば $u_t = u_1^3 u_3$ の recursion operator

$$\begin{aligned} \mathcal{D} &= u_1^2 D^2 - u_1 u_2 D + 3u_1 u_3 + u_2^2 \\ &\quad - 2D^{-1} \cdot (u_1 u_4) - 4D^{-1} \cdot (u_2 u_3) + 2u_1^3 u_3 D^{-1} \cdot \left(\frac{u_2}{u_1^3}\right) \end{aligned}$$

にたいして $\text{herop}(\mathcal{D})$ を計算すると 210 項にもなってしまう視察による手計算では限界がある。そこで少し泥縄だが $m > 0$ なら任意の有理式 p にたいして $\text{ind}(f(m) * p)$ を $f(m-1) * p - \text{ind}(f(m-1) * df(p, x))(g)$ (についても同様) とする部分積分の procedure pintfg を作ったところ、知られている recursion operator にたいして herop を適用した結果はこの部分積分の procedure pintfg によってすべて 0 となった。

D. Schouten bracket を計算する。

今までに説明した procedure を用いると、二つの微分積分作用素 \mathcal{H} と \mathcal{K} にたいして (1.12) の右辺 (正確にいうと (1.12) の右辺で積分してないもの) を計算する procedure $\text{scbra}(\mathcal{H}, \mathcal{K})$ を作ることが出来る。但しオペレーター f, g, h は大域的に宣言される。 \mathcal{H} が Hamilton 作用素であることをみるには、 $\text{scbra}(\mathcal{H}, \mathcal{H})$ を代数的に積分して 0 となるかどうか、すなわち $\text{scbra}(\mathcal{H}, \mathcal{H})$ がある関数の微分で書けるかどうかを調べなければならない。そのためには部分積分の考えが有効である。容易にわかるように $fD(g) = D(fg) - D(f)g$ または $D^{-1}(f)D(g) = D(D^{-1}(f)(g)) - fg$ 等の部分積分の公式が成り立つ。この部分節では上記の公式は (D -image を除いて考えるから) $fD(g) = -D(f)g$ または $D^{-1}(f)D(g) = -fg$ となって簡単になる。しかし定義式 (1.12) からわかるように、一般的に言って scbra の計算結果は極めて長い式になる。例えば Hamilton 作用素

$$\begin{aligned} \mathcal{H} &= u^5 D^3 + \frac{15}{2} u^4 u_1 D^2 + \left(\frac{15}{2} u^4 u_2 + \frac{49}{4} u^3 u_1^2\right) D + \frac{5}{2} u^4 u_3 + \frac{49}{4} u^3 u_1 u_2 + \frac{27}{8} u^2 u_1^3 \\ &\quad + \frac{1}{2} u_1 D^{-1} \cdot (u^3 u_3) + \frac{3}{4} u_1 D^{-1} \cdot (u^2 u_1 u_2) + \frac{1}{2} u^3 u_1 u_2 + \frac{3}{4} u^2 u_1 u_2 \end{aligned}$$

にたいして $\text{scbra}(\mathcal{H}, \mathcal{H})$ を計算すると 869 項にもなりプリンターから打ち出すとその出力は数十ページにもわたる。このような複雑な式を簡単化するため以下の三つの procedure mvind , pintind , pint を作った。

(1) Procedure $\text{mvind}(p, \text{list})$; p は簡単化されるべき式で list はいくつかの式からなる Algebraic mode のリストである。この procedure は p のなかに $f * \text{ind}(g)$ という項があると g がリスト list に含まれていない場合には $-g * \text{ind}(f)$ とし、含まれている場合は何もしない。この procedure によって例えば $u(1) * f(0)$ が list に含まれているとして

$u(3)*u(0)**3*g(1)*h(0)*ind(u(1)*f(0))+u(1)*f(0)*ind(u(3)*u(0)**3*g(1)*h(0))$ は 0 になる。

(2) Procedure $pintind(p, list)$; (1) と同様に p は簡単化されるべき式で $list$ は式からなる Algebraic mode でのリストである。この procedure は $list$ に含まれている各々の式 g にたいして p における $ind(g)$ の係数 f をひろい、その係数 f を積分したものを h と置くととき (すなわち $D(h) = f$ である)、 p の $f*ind(g)$ の部分を $-h*g$ で置き換える。例えば $u(1)*f(0)$ が $list$ に含まれているならば

$$(u(4)*u(0)**3+3*u(3)*u(1)*u(0)**2)*g(0)*h(0)*ind(u(1)*f(0)) \\ +u(3)*u(0)**3*(g(1)*h(0)+g(0)*h(1))*ind(u(1)*f(0))$$

は $pintind$ によって $-u(3)*u(1)*u(0)**3*f(0)*g(0)*h(0)$ となる。式 f が積分出来なければこの procedure は実行できない。積分のアルゴリズムについては不完全ではあるが次項で述べる。

(3) Procedure $pint(p)$; この procedure は以下の二つの procedure からなる。

(I) Procedure $pintfgh(p)$; 前部分節で作った $pintfg$ と同様にして部分積分を用いて有理式 p に含まれるオペレーター f, g, h の引数を減らす。

(II) Procedure $pintu$; (i): まず p に含まれる $u(j)$ のなかで j が最大となるものを $u(m)$ とする。 m が 0 であるか、 p が $u(m)$ に関して二次以上なら p を返しそうでなければ次のステップ (ii) へ。(ii): p における $u(m)$ の係数を q とするとき q が $u(m-1)$ を含まなければ $p := p - D(u(m-1)q)$ とし、 q が $u(m-1)$ を含むときは q の $u(m-1)$ に関する次数を k 、 $u(m-1)^k$ の係数を r として $p := p - D(\frac{1}{k+1}u(m-1)^{k+1}r)$ とする。更に上記 (i), (ii) の手続きを $m=0$ となるか p の $u(m)$ に関する次数が二次以上になるまで繰り返す。

(2) の $pintind$ における係数の積分に際しては上記 $pintu$ で用いたものと同じアルゴリズムを用いる。

文献 [W] で得られた幾つかの歪対称な微分積分作用素について上記の三つの部分積分の procedure を適当に組み合わせることによって、それらが Hamilton 作用素であること、またそれらの幾つかの組が Hamilton 対となることを確かめることが出来る。その際、三つの procedure の適用順序によって随分計算時間が異なることを注意する。上記の部分積分のアルゴリズムはかなり有用ではあるがまだ不完全な点が多い。とりわけ $mvind, pintind$ の第二引数の $list$ としてどのような式の組み合わせをとればよいかという点については、全くの視察によるだけで自動化からは程遠い。

REFERENCES

- [F] B. Fuchssteiner, *Application of hereditary symmetries to nonlinear evolution equations*, Non-linear Anal., Theory, Method and Appl. **3(6)** (1979), 849-862.
- [FF] B. Fuchssteiner and A.S. Fokas, *Symplectic structures, their Bäcklund transformations and hereditary symmetries*, Physica **4D** (1981), 47-66.
- [FW1] A. Fujimoto and Y. Watanabe, *Polynomial evolution equations of not normal type admitting nontrivial symmetries*, Phys. Lett. **136A(6)** (1989), 294-299.

- [FW2] A. Fujimoto and Y. Watanabe, *Classification of the third-order polynomial evolution equations of not normal type admitting nontrivial symmetries*, Memoirs Fac. Eng. Hiroshima Univ. **10(2)** (1989), 23–33.
- [GD1] I.M. Gel'fand and I.Ya Dorfman, *Hamiltonian operators and algebraic structures related to them*, Funct. Anal. Appl. **13(4)** (1979), 248–262.
- [GD2] I.M. Gel'fand and I.Ya Dorfman, *The Schouten bracket and Hamiltonian operators*, Funct. Anal. Appl. **14(3)** (1980), 223–226.
- [O] P.J. Olver, *Evolution equations possessing infinitely many symmetries*, J. Math. Phys. **18(6)** (1977), 1212–1215.
- [W] Y. Watanabe, *Hamiltonian structure and formal complete integrability of third-order evolution equations of not normal type*, in K. Shiohama (ed.) *Geometry of Manifolds*, Academic Press Inc., 1989, 29–38.